Do it yourself – automated thin film design using genetic algorithms

W.Theiss, W.Theiss Hard- and Software, Aachen, Germany

Abstract

The selection of an appropriate thin film stack as solution for a given optical design problem requires experience. First the materials have to be selected and put into the right sequence, then the layer thicknesses have to be optimized. Whereas the final thickness adjustments can easily be done by a computer, material selection is usually 'man made'. If a thin film expert is not available for some reason the development of the coating is blocked. One can overcome this problem applying a genetic algorithm to thin film design: In many cases it is sufficient to state the optimization problem and let the algorithm develop the solution in an evolutionary process. The principle of the method is explained, and its strength and limitations are discussed. Typical industrial thin film stacks (architectural glass coatings, AR coatings, optical filters) are used to demonstrate the application of the method.

Introduction

Many technical devices are based on the properties of thin films. In order to optimize the device performance the thin film stack needs to be carefully designed with respect to both choice of materials and layer thicknesses. For a given technical problem, the consultation of a thin film expert will usually lead to a satisfying and well-approved solution.

If a quick solution is needed with no time for contacting an experienced thin film designer, or if there is not enough funding for doing so, the "problem owner" itself has to do the optimization. This may be a difficult task if the required expertise is not available. The method described in this article may help in such a situation. The optimization problem is reduced to the formulation of the goal, specifying a list of available thin film materials, and the number of possible deposition steps (which equals the number of possible layers).

The genetic algorithm described below tries to "grow" a good solution for the design problem. Starting with many arbitrary layer stacks in the first generation, the properties of the better ones are taken over and mixed in the next generation. The worst layer stacks are eliminated. Doing this for many generations with some additional randomness (mutation) may lead to very good solutions as the examples in this article will show.

After initial setup the method works completely automatic and can solve problems overnight. It seems like a big disadvantage that the evolution spends almost all the time processing layer stacks which would be immediately rated as "stupid" by an expert. On the other hand, the genetic algorithm generates solutions in a completely different way than any human expert, and this gives the chance to find designs which no one would think of – at least theoretically.

After a description of the method we first discuss simple problems with a few layers only, and then show how genetics can be efficiently used to design optical filters involving many layers.

Turning a thin film design problem into an evolutionary process

Genetic algorithms (sometimes also called 'evolutionary' algorithms) can be used for all kinds of problems [1]. The scheme is always the same: One starts with a large population of individuals who are characterized by their individual genes. For each member of the population its fitness is computed. Those individuals with a high fitness go on to the next generation, the ones with the lowest fitness values are taken out. The missing places are filled up with children of the fitter members. Children are generated by mixing the genes of two parents. It turned out to be useful to add a little bit "thunder and lightning" which means that some genes of the next generation are changed by random mutations.

This general description makes clear what we have to do in order to apply this method to optical thin film design: There must be a fitness number characterizing the quality of a layer stack, and we must be able to represent a layer stack as a sequence of genes.

Test case

The practical application of a genetic method is explained in detail using a simple thin film problem. If you want to heat car glass in order to remove ice you need to deposit an electrically conductive layer. On the other hand you still have to be able to look through the window – the transmission should not be reduced too much by the coating. We will show in the following how the recipe for such a coating can be found using an evolutionary algorithm.

Reducing the problem to the optimization of a single quantity (fitness)

This step is very easy because in thin film optimization the target is usually to minimize the deviation between wanted properties and actual properties of the layer stack. The inverse of the deviation can be taken as the fitness of the layer stack.

In our test case we have to balance electrical and optical performance. A good number for the electrical properties is the sheet resistance – here we ask for a value of 2.2 Ohm which seems to be sufficient for melting ice on a car window. The optical part is handled by the light transmittance averaged in the visible applying human sensitivity. This quantity is called VLT [2] and we ask for its maximum value which is 1.

If the sheet resistance of a coating is called SR and the transmittance is labeled as VLT, the quantity to be minimized is

$$Deviation = \frac{1}{10}(SR - 2.2)^2 + (VLT - 1)^2$$

where it was decided to scale down the importance of the correct sheet resistance by a factor of 10, i.e. we want definitely a coating with high optical transmission.

The fitness of a layer stack is expressed as

$$Fitness = \frac{1}{0.0001 + Deviation}$$

which avoids numerical problems if the deviation becomes zero.

Convert layer stacks into genes

How can be represent layer stacks as genes? The method described here implements the following scheme: The total stack is produced by a sequence of deposition steps, each of which generates one layer of a certain material with a certain mechanical thickness. Both the thickness and the choice of material are represented using a 64 bit floating point number as explained below. Hence every thin film is defined by 128 bit which are the genes of that layer. Linking the genes of the individual layers a stack of N layers is completely characterized by a chromosome of N*128 bit.

We decided to map each deposition step into a bit pattern by working with a finite range of possible thickness values. For each material we specify a minimum and a maximum thickness that can be applied in a single deposition step. This way we can easily represent the sequence of sputtering devices in an inline coater: For each cathode you have the option to mount various targets (=materials) with limited sputtering rates, i.e. limited thickness value. Having this in mind, we use the expression

$$d = d_{min} + \frac{d_{max} - d_{min}}{1 + |f|}$$

where f is a floating point number with 64 bit (double). Every choice of f leads to a thickness between d_{min} and d_{max}. Table 1 shows the 9 materials and thickness ranges that have been used as building blocks in this example.

The choice of material for each layer is done in a similar way: The index N of the material in the list is obtained by the expression

$$N = truncate\left(1 + \frac{N_{Max} - 1}{1 + |g|}\right)$$

where g (like f before) is a 64 bit floating number. Putting the bits of f and g behind each other leads to a sequence of 128 bits (= genes) telling the evolution what to do:

$$01000111010010001101101 \ldots 01001101100000110110 \;=\; build\ a\ layer\ of\ 13.784\, nm\ SiN$$

Run the evolution

The method begins by creating many individuals with random genes in the first generation (sometimes called seed generation). For each member the fitness is computed, and a ranking sorts the individuals by their fitness. Then the evolution is started: The best individuals move on to the next generation without change (in our implementation the best and the second best). The other members of the next generation are 'born' as children of two parents in the previous generation. The probability of being selected as 'parent' depends on the ranking, with higher chances for higher ranked individuals. The genes of a child are built by copying gene sequences of mother and father – in our genetic algorithm the copying action jumps from mother to father genes and back at two random positions in the chromosome.

Now, between 2 generations, a 'mutation' step is inserted which introduces random modifications of genes which occur in nature as well (e.g. by cosmic radiation or errors during gene reproduction). Besides bad effects on the individual, mutations may have a positive effect on the development of a good solution. In our algorithm mutation means to swap 2 arbitrarily selected genes of the chosen individual.

Now the next generation is processed: Fitness values are computed and the ranking is obtained. After an initially fixed number of generations the algorithm stops and one can inspect the best solution. It is generally accepted that one should do several evolution runs for a given problem. Only in very simple cases will the results of all runs agree – most of the time different runs end with different solutions.

Fig. 1 shows solutions obtained by the genetic algorithm for coatings with 1, 2 and 3 layers. The 3-layer system is a very satisfying solution for the given problem. Note that no knowledge of thin film design has been involved in order to find it – the ingredients of the solution have been entering the design target and knowing the optical constants of all candidate materials.

Questions concerning thin film patents

Using this problem to demonstrate the evolutionary algorithm has been inspired by a patent [3]. The patent states the same problem with similar solutions. This raises several questions: Is it fair to grant patents for the solution of an optimization problem? After running an optimization by a computer, is the first human reader of the result the patent holder? Or the person who pushed the 'Start' button? Or the programmer of the algorithm?

Refinement of the best solution

Simple cases with a few layers only usually do not need any refinement after the evolutionary optimization. In cases involving a large number of layers, however, we found that the result of the genetic method could be improved significantly by running a downhill optimization, using the thickness results and the choice of materials found by the evolution as starting configuration. The examples of optical filter design below show how important this refinement step can be.

Examples

Low-e glass coatings

The 'heatable car glass' problem is almost the same as the design problem of low-emission coatings for architectural glass. The genetic algorithm easily 'invents' single silver low-e coatings as well as double silver and triple silver stacks if more than 3 layers are allowed in the design.

AR coatings

We have successfully used the evolutionary method described above to design AR coatings for silicon detectors and the optimization of AR coatings for solar cells. Designing AR coatings one must be careful in defining the true goal of the optimization: If the target is simply defined as 'zero reflectance' the optimization may end up in very good solutions involving metallic layers which absorb part of the radiation. If the real goal is a high transmission then these solutions are bad. To avoid this 'misunderstanding' you should define a high transmission as target of the optimization. A low reflectance will automatically be the result of good AR designs.

Optical filters

The design of optical filters involves many layers and is a little bit more challenging. We start with the task to develop an 'edge filter', i.e. a transmission filter on glass that has a sharp transition from high to zero transmittance at a certain wavelength. Our first approach is to use 10 layers and the material list of table 1. The result is shown in fig.2 – it is not very convincing since the coating fails to generate transmittance values close to zero where they are wanted.

In order to obtain better results it helps to learn a little bit about optical filters, without getting very deep into it (although that would also be a good idea, see [4][5]). On public sources like [6] we can learn that important building blocks are repeated double layers with high and low refractive index materials. These so-called Bragg reflectors may generate very high reflectance values that go along with low transmission on the other hand, and that's what we need. Our next approach is to use 6 layers (A,B,C,D,E,F) where the two in the middle are repeated 20 times. The layer stack then reads $AB(CD)^{20}EF$ where the index 20 means to repeat the enclosed substack 20 times – hence the total number of layers in the stack is 44. Note that the algorithm is still free to assign materials and thickness values to the 6 layer definitions (A,B,C,D,E,F). With this help a sharp edge is found without problem (fig. 2, left). However, there are still unwanted interference patterns in the transparent region of the filter below 560 nm.

These can be avoided by stepping back from the regular repetition of the double layer: Take the result of the genetic algorithm, remove the synchronization of the 20 double layers and fit all 44 layer thicknesses of the stack individually (using a downhill method) gives a very satisfying result (fig. 2, right). This refinement step after the evolution seems to be essential to achieve good results for designs involving many layers.

Finally fig.3 shows the successful design of a rectangular bandpass filter involving 64 layers (based on 2 Bragg reflectors with a spacer layer in between) and a 19 layer filter that mimics the sensitivity of the human eye in transmission. In both cases the refinement step discussed above has been applied to the best result of the genetic algorithm.

Conclusion

It was shown how the general scheme of evolutionary algorithms can be adapted to be used for optical thin film design, including automatic material selection. The problem reduces to the definition of the design target and some rather basic decisions concerning the structure of the layer stack. The author confirms that the satisfying results have been obtained without any significant knowledge on thin film design – the layer stack populations have grown to this level of perfection by themselves.

References

[1] Mitchell, Melanie, "An Introduction to Genetic Algorithms", The MIT Press Cambridge, Massachusetts, 1999

[2] NFRC 300-2004

[3] Patent specification 1307642, Patent Office, 25 Southampton Buildings, London, WC2A 1AY, 1973

[4] Macleod, H. Angus, "Thin-Film Optical Filters, Fourth Edition (Series in Optics and Optoelectronics)", CRC Press (2010), ISBN-13: 978-1420073027

[5] Willey, Ronald R., "Practical Design and Production of Optical Thin Films", CRC Press (2002), ISBN 9780824708498

[6] schott_interference_filters_descript_2013_eng.pdf, download at http://www.schott.com/advanced_optics/english/download/

Material	d_{min} [nm]	d_{max} [nm]	*Material index*
Ag	3	30	1
Au	3	30	2
Cu	3	30	3
ITO	20	300	4
SiN	5	100	5
SiO2	5	100	6
Ta2O5	5	100	7
TiO2 (cryst.)	5	25	8
TiO2 (amorph.)	5	25	9

Table 1: Materials and thickness ranges for the 'heated car glass' problem

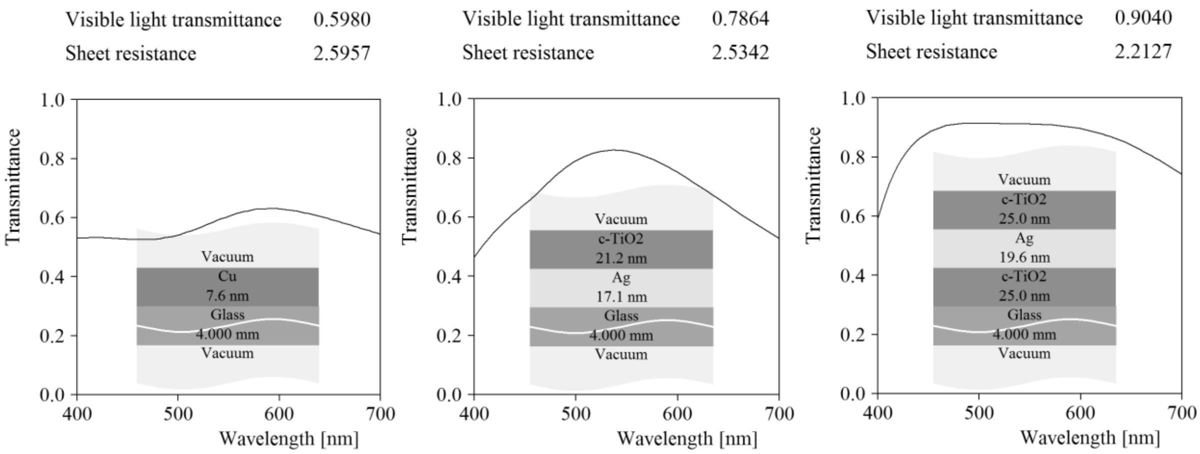

Fig. 1: Solutions for the 'heatable car glass' problem with 1, 2 and 3 layers. A population of 100 individuals over 50 generations has been followed. On a state-of-the-art laptop this takes less than one minute.

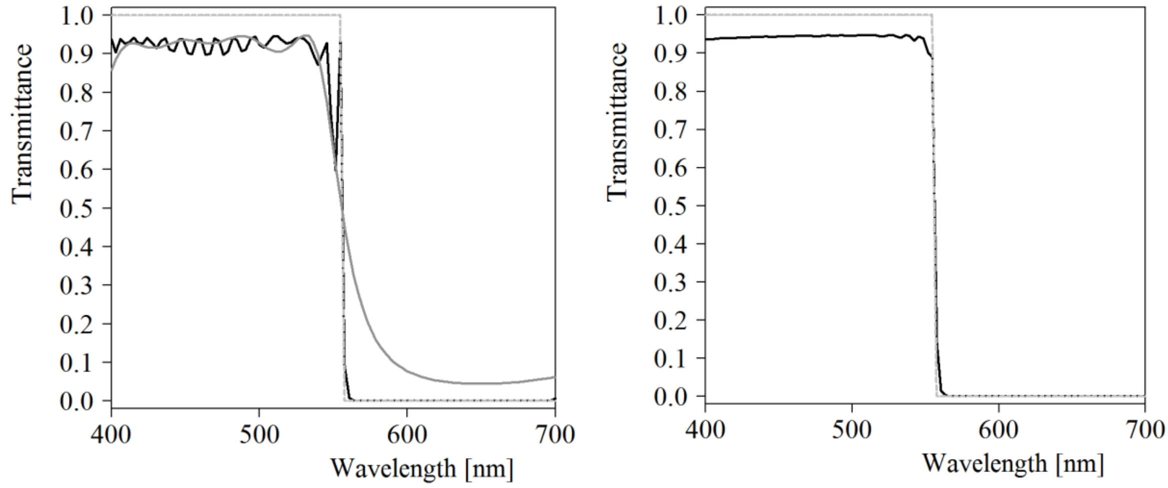

Fig. 2: ‘Edge filter: The design attempt using 10 layers (left, gray, solid) does not reach the low transmittance of the target spectrum (dashed) above 560 nm. The design using a Bragg reflector (left, black, solid) is much better but has annoying structures in the transparent region. The graph on the right shows the final result after refinement (see text). Note that a transmittance of 1 cannot be reached because of the backside reflection of the glass – this could be avoided applying an AR coating on the backside.

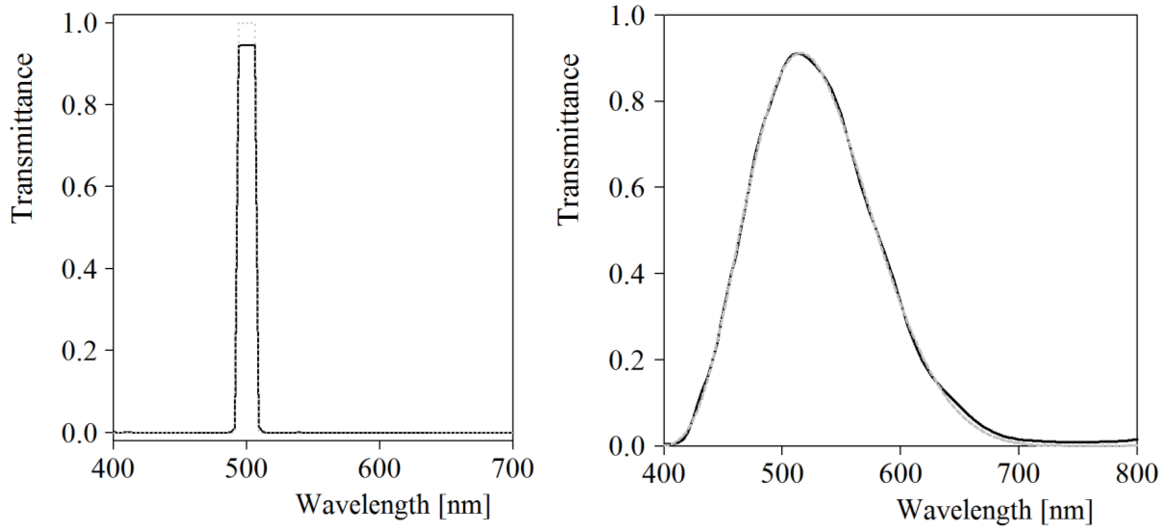

Fig. 3: Bandpass filter, consisting of 64 layers, and a transmission filter with 19 layers that reproduces the sensitivity of the human eye. The target spectra are shown in dashed gray, the design spectra in solid black.